ГЛАВА XA.
Кривые в машинном обучении

…нужно оценивать не только по делам,
но и по стремлениям.
Демокрит Абдерский

Таланту только в счастливые минуты удается составить из точек линию, которую гений проводит одним росчерком пера.
Мария фон Эбнер-Эшенбах

Продолжаем рассматривать задачу бинарной классификации с выборкой

$$(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m),\ y_1, \ldots, y_m \in \{0, 1\},$$

будем оценивать качество алгоритма, который получил оценки принадлежности к классу 1 для всех объектов выборки:

$$b_1, b_2, \ldots, b_m \in \mathbb{R}. \qquad \text{(XA.1)}$$

Можно считать, что оценки принадлежат отрезку $[0,1]$ (хотя в этой главе это будет не обязательно, как и интерпретация оценок как вероятностей). Окончательные ответы алгоритма могут получаться бинаризацией с помощью порогового решающего правила:

$$\forall i \in \{1, 2, \ldots, m\} \quad a(x_i) = I[b(x_i) \geq \theta] = \begin{bmatrix} 1, & b(x_i) \geq \theta, \\ 0, & b(x_i) < \theta, \end{bmatrix}$$

(если оценка выше фиксированного порога $\theta \in \mathbb{R}$, то объект относим к классу 1, иначе – к классу 0).

Площадь под ROC-кривой

Часто результат работы алгоритма на фиксированной выборке визуализируют с помощью **ROC-кривой**[1] **(Receiver Operating Characteristic)**, а качество

[1] Иногда говорят «кривая ошибок», хотя это не очень удачный термин. Про кривую см. Tom Fawcett An introduction to ROC analysis // Pattern Recognition Letters Volume 27 Issue 8, 2006, P. 861-874.

оценивают как **площадь под этой кривой** $\mathrm{AUC}_{\mathrm{ROC}}$ **– AUC** (**Area Under the Curve**). Для начала покажем на конкретном примере, как строится ROC-кривая.

Пусть алгоритм получил оценки, показанные в табл. 1 (там также показаны истинные метки объектов), упорядочим строки таблицы по невозрастанию оценок – получим табл. 2. Ясно, что в идеале её столбец «класс» тоже станет упорядочен (сначала будут идти 1, потом 0); в самом худшем случае – порядок будет обратный (сначала 0, потом 1); в случае «случайных ответов[1]» будет случайное распределение 0 и 1 в столбце.

Чтобы нарисовать ROC-кривую, надо взять единичный квадрат на координатной плоскости с вершинами в $\{0,1\}\times\{0,1\}$, см. рис. XA.1, разбить его на m_1 равных частей горизонтальными линиями и на m_0 – вертикальными, где m_1 – число 1 среди правильных меток выборки, для которой строится кривая (в нашем примере $m_1 = 3$), m_0 – число нулей ($m_0 = 4$). В результате квадрат разбивается сеткой на $m_1 \times m_0$ прямоугольников. Вершины сетки имеют координаты вида $(i/m_0,\ j/m_1)$, $i \in \{1,2,\ldots,m_0\}$, $j \in \{1,2,\ldots,m_1\}$.

id	оценка (b)	класс
1	0.5	0
2	0.1	0
3	0.2	0
4	0.6	1
5	0.2	1
6	0.3	1
7	0.0	0

Табл. 1

id	оценка (b)	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

id	> 0.25	класс
4	1	1
1	1	0
6	1	1
3	0	0
5	0	1
2	0	0
7	0	0

Табл. 3

Теперь будем просматривать строки табл. 2 сверху вниз и прорисовывать на сетке отрезки, переходя из одного узла в другой. Стартуем из точки $(0,0)$. Находясь в точке $(i/m_0,\ j/m_1)$, берём очередную группу строк с одинаковой меткой (следующей по убыванию), если в группе k_0 объектов с меткой 1 и k_1 объектов с меткой 0, то «переходим» в точку $((i+k_0)/m_0, (j+k_1)/m_1)$, т.е. соединяем точки $(i/m_0,\ j/m_1)$ и $((i+k_0)/m_0, (j+k_1)/m_1)$ отрезком. В итоге (после просмотра всей таблицы) приходим в точку $(1,1)$, т.к. сделаем в сумме m_1 шагов вверх длины $1/m_1$ и m_0 шагов вправо длины $1/m_0$.

[1] Когда оценка, порождённая моделью, случайна и не зависит от описания объекта.

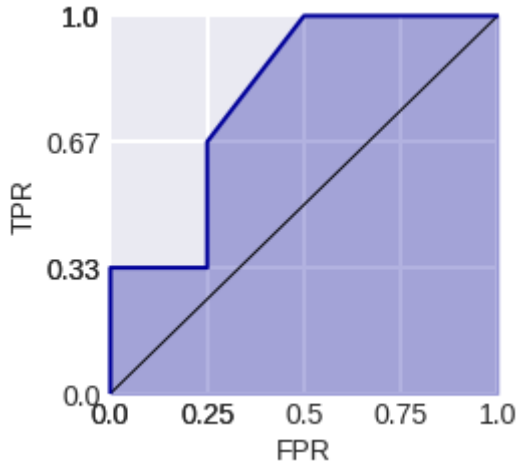


Рис. ХА.1. Пример ROC-кривой.

ROC-кривая для конечной выборки при попарно разных оценках будет ступенчатой функцией.

На рис. ХА.1 показан путь для нашего примера – это и является ROC-кривой. Если оценки всех объектов попарно различны, то для построения кривой просто идём по строкам отсортированной табл. сверху вниз, когда в строке метка 1 делаем шаг вверх длины $1/m_1$, когда метка 0 – шаг вправо длины $1/m_0$. Понятно, что для константного алгоритма (который для всех объектов выдаёт одну и ту же оценку) ROC-кривая является диагональю квадрата: отрезком, который соединяет точки $(0,0)$ и $(1,1)$ (у нас одна группа точек с одинаковой оценкой и мы на первом шаге переходим из начальной точки в конечную). Поэтому ниже мы не будем задаваться вопросом об оптимальном константном алгоритме (для всех константных алгоритмов ROC-кривая одинакова).

AUC_{ROC} (также называют AUC ROC или ROC AUC) – площадь под ROC-кривой – часто используют для оценивания качества упорядочивания[1] алгоритмом объектов двух классов. Ясно, что это значение лежит на отрезке $[0,1]$. В нашем примере

$$\text{AUC}_{\text{ROC}} = 9.5/12 \approx 0.79.$$

Выше мы описали варианты идеального, наихудшего и случайного следования меток в упорядоченной таблице. Идеальному соответствует ROC-кривая, проходящая через точку (0, 1), площадь под ней равна 1 (рис. ХА.2 слева), т.к. при построении мы будем шагать вверх, пока не закончатся объекты с метками 1, а потом вправо. Наихудшему – ROC-кривая, проходящая через точку (1, 0), площадь под ней – 0 (рис. ХА.2 справа), сначала шагаем вправо, пока не

[1] т.к. конкрентые значения оценок не важны, а важен порядок объектов в табл. 2.

закончатся объекты с меткой 0. Случайному – что-то похожее на диагональ квадрата, площадь примерно равна 0.5 (см. рис. XA.2, посередине).

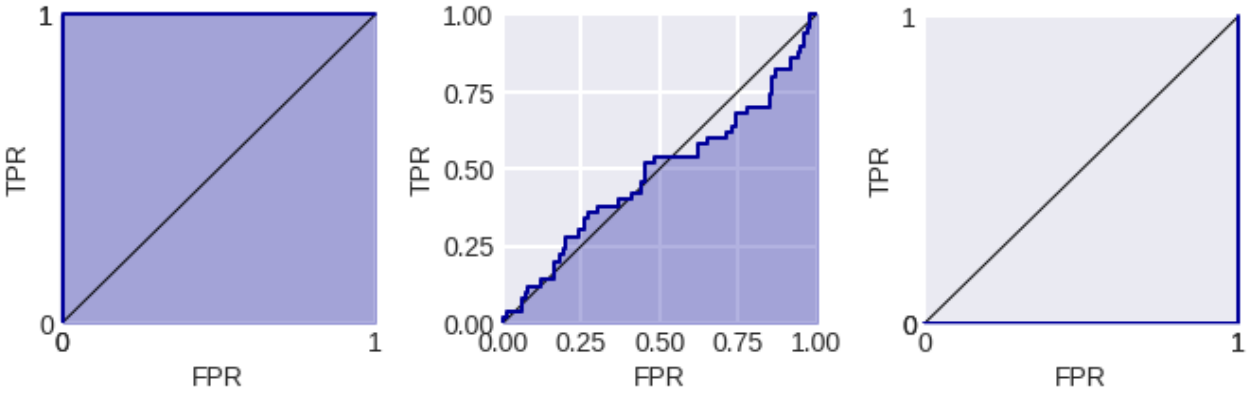


Рис. XA.2. ROC-кривые для наилучшего ($AUC_{ROC} = 1$), случайного ($AUC_{ROC} \approx 0.5$) и наихудшего ($AUC_{ROC} = 0$) алгоритма.

ROC-кривая считается неопределённой для выборки, целиком состоящей из объектов только одного класса. Большинство современных реализаций выдают ошибку при попытке построить её в этом случае.

ROC_AUC((0,0,0), (0.2, 0.5, 0.6)) = NaN

Заметим, что в сетке из $m_1 \times m_0$ прямоугольников каждому прямоугольнику взаимно однозначно можно сопоставить пару объектов из разных классов. На рис. XA.3 горизонтальным и вертикальным блокам сетки приписаны номера (id) объектов: снизу вверх id = 4, 6, 5 они соответствуют объектам с меткой 1, в порядке невозрастания оценки, слева направо – объектам с меткой 0. При построении ROC-кривой, когда мы анализируем группу объектов с одинаковой меткой, то перемещаемся по участку квадрата, который помечен номерами соответствующих объектов. Если закрасить область под кривой, то полностью закрашенные прямоугольники будут соответствовать парам объектов разных классов[1], которые алгоритм верно упорядочил: объект с меткой 1 имеет большую оценку, чем объект с меткой 0. Например, все пары с объектом id=4 верно упорядочены (ведь к него самая большая оценка) и нижний участок сетки полностью закрашен. Полностью незакрашенные прямоугольники соответсвуют неверно упорядоченным парам, например id=6 и id=1:

$$b_6 = 0.3 < b_1 = 0.5,$$

$$y_6 = 1 > y_1 = 0.$$

[1] Строго говоря, это не всегда верно. Например, неверно для константного алгоритма. Но понятно, как перерисовать раскраску на сетке, чтобы это свойство выполнялось и площадь закраски совпадала с ROC AUC.

На рис. ХА.1 наполовину закрашенный прямоугольник соответствует паре объектов с одинаковой оценкой, но с разными метками:

$$b_2 = b_4 = 0.2,$$

$$y_2 = 0 < y_4 = 1.$$

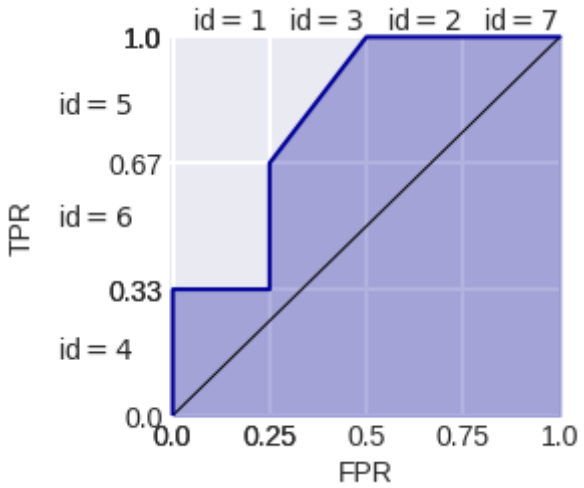


Рис. ХА.3. Каждый прямоугольник (ячейка сетки) соответствует паре объектов.

Таким образом, **показатель $\mathbf{AUC_{ROC}}$ равен доле пар объектов вида (объект класса 1, объект класса 0), которые алгоритм верно упорядочил**, т.е. первый объект идёт в упорядоченном по невозрастанию оценок списке раньше. Численно это можно записать так:

$$\mathrm{AUC}_{\mathrm{ROC}} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{m} I[y_i < y_j] \cdot I'[b_i < b_j]}{\sum_{i=1}^{m}\sum_{j=1}^{m} I[y_i < y_j]}, \qquad \text{(ХА.2)}$$

$$I'[b_i < b_j] = \begin{cases} 0, & b_i > b_j, \\ 0.5 & b_i = b_j, \\ 1, & b_i < b_j, \end{cases}$$

$$I[y_i < y_j] = \begin{cases} 0, & y_i \geq y_j, \\ 1, & y_i < y_j, \end{cases}$$

здесь b_i – оценка i-го объекта (получена алгоритмом), y_i – его метка (класс), m – число объектов в выборке. Формула (ХА.2) хороша тем, что легко обобщается и на другие задачи обучения с

Если индикатор не переопределить так, чтобы он принимал значение 0.5, то формула буде неверна.

учителем, поскольку тут используется лишь сравнение меток и оценок. Вероятностная интерпретация показателя AUC_{ROC} понятна, он является **оценкой вероятности правильного упорядочивания случайной пары объектов из разных классов**. Заметим, что такой смысл популярного показателя качества сложно объяснить непрофессионалу[1].

Пока наш алгоритм выдавал оценки (XA.1) принадлежности к классу 1. На практике нам часто надо будет решить: какие объекты отнести к классу 1, а какие к классу 0. Для этого нужно выбрать некоторый порог бинаризации $c \in \mathbb{R}$ (объекты с оценками выше порога считаем принадлежащими классу 1, остальные – 0): $a(x) = I[b(x) \geq c]$, где $a(x)$ – ответ алгоритма (метка) на объекте x, $b(x)$ – оценка принадлежности к классу 1, полученная алгоритмом. Выбору порога соответствует выбор точки на ROC-кривой (узел сетки). Например, для порога $c = 0.25$ и нашего примера – точка указана на рис. XA.4 (1/4, 2/3). см. табл. 3.

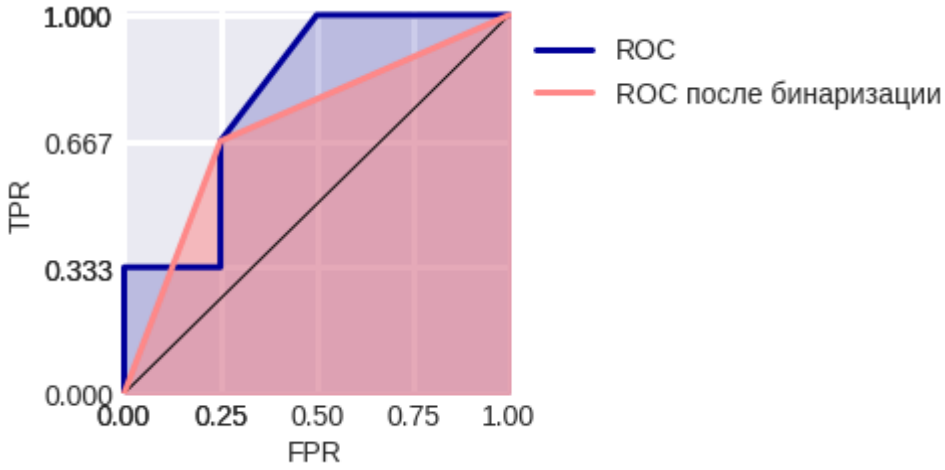


Рис. XA.4. Выбор порога для бинаризации.

Заметим, что 1/4 – это доля точек класса 0, которые неверно классифицированы нашим алгоритмом, это называется **FPR = False Positive Rate**:

$$\text{FPR} = \frac{\sum_{i=1}^{m} I[a_i = 1] I[y_i = 0]}{\sum_{i=1}^{m} I[y_i = 0]} \qquad \text{(XA.3)}$$

[1] Например, когда Accuracy=0.9, то это интерпретируют так: «в 90% случаях ответ алгоритма будет верным». Когда ROC AUC = 0.9, то интерпретация не такая простая: «если взять случайный объект из класса 1 и случайный объект из класса 0, то с вероятностью 0.9 у первого объекта будет выше оценка»...

в этой формуле $a_i = a(x_i)$ – ответ (метка) алгоритма на i-м объекте выборки, т.е. после бинаризации, в числителе FP (False Positive) – число объектов из класса 0, которые были отнесены к классу 1, в знаменателе m_0 – число объектов класса 0. Аналогично, 2/3 – доля точек класса 1, которые верно классифицированы нашим алгоритмом, это называется **TPR = True Positive Rate**:

$$\text{TPR} = \frac{\sum_{i=1}^{m} I[a_i = 1] I[y_i = 1]}{\sum_{i=1}^{m} I[y_i = 1]}, \qquad \text{(XA.4)}$$

в числителе TP (True Positive) – число объектов верно отнесённых к классу 1, в знаменателе m_1 – число объектов класса 1. Именно в этих координатах (FPR, TPR) построена ROC-кривая. Часто в литературе её и определяют как **кривую зависимости TPR от FPR при варьировании порога для бинаризации**.

Кстати, если ответы бинаризованного алгоритма снова интерпретировать как оценки, то можно и для них построить ROC-кривую, но она довольно просто устроена, см рис. XA.4 (состоит не более чем из трёх точек, соединённых отрезками). Интересно, что площадь под такой кривой равна сбалансированной точности (balanced accuracy) бинаризованного ответа.

Площадь под ROC-кривой не зависит от баланса классов, в том смысле, что если какой-то класс «проредить» (удалить из него случайное множество объектов), то значение функционала сильно не изменится, поскольку не сильно изменятся FPR (XA.3) и TPR (XA.4), а значит и сама ROC-кривая. Действительно, после удаления случайного множества объектов класса доля объектов класса с оценкой выше определённого порога не должна сильно измениться.

Формулы (XA.3), (XA.4) позволяют обобщать ROC-кривую и площадь под ней на случай наличия весов w_i у объектов, для этого надо перейти к весовым FPR и TPR, например

$$\text{wTPR} = \frac{\sum_{i=1}^{m} w_i I[a_i = 1] I[y_i = 1]}{\sum_{i=1}^{m} w_i I[y_i = 1]}.$$

Геометрическая интерпретация весовой ROC-кривой будет в использовании неравномерной сетки: стороны прямоугольника (ячейки сетки), который соответствует i-му объекту из класса 1 и j-му объекту из класса 0 равны

$$\frac{w_i}{\sum_{t=1}^{m} w_t I[y_t = 1]}, \frac{w_j}{\sum_{t=1}^{m} w_t I[y_t = 0]},$$

т.е. его площадь будет пропорциональна произведению весов $w_i w_j$.

Оптимизировать площадь под ROC-кривой напрямую затруднительно по нескольким причинам:

- эта функция недифференцируема по параметрам модели,
- она в явном виде не разбивается на отдельные слагаемые, которые зависят от ответа только на одном объекте (как, например, происходит в случае log_loss).

Общие подходы к оптимизации следующие:

- замена в (XA.2) индикаторной функции на похожую дифференцируемую функцию (например, сигмоиду);
- использование смысла функционала: если это вероятность верного упорядочивания пары объектов, то можно перейти к новой выборке, состоящей из пар, и построить классификатор для таких пар (правильно или нет они упорядочены) –

  $$\{(x_i, y_i)\}_{i=1}^{m} \quad \rightarrow \quad \{((x_i, x_j), t_{ij})\}_{ij \in I},$$

  где $I = \{(i, j) \mid y_i \neq y_j\}$,

  $$t_{ij} = \begin{cases} 1, & y_i = 0, y_j = 1, \\ 0, & y_i = 1, y_j = 0. \end{cases}$$

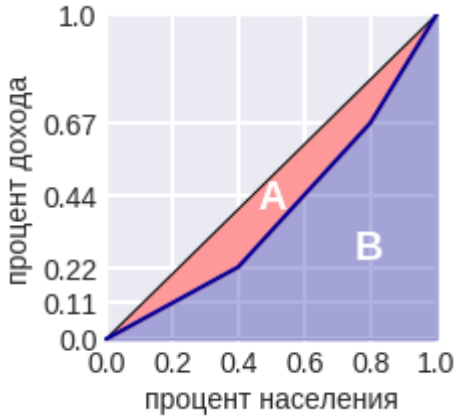
- при ансамблировании нескольких алгоритмов использовать предварительное преобразование их оценок в ранги[1] (логика здесь простая: AUC_{ROC} зависит только от порядка объектов, поэтому конкретные оценки не должны существенно влиять на ответ).

[1] Есть такая функция: pandas.DataFrame.rank.

По поводу использования ранговой нормировки заметим, что **показатель AUC ROC не зависит от строго возрастающего преобразования оценок** (например, возведения неотрицательных оценок в квадрат), поскольку зависит не от самих оценок, а от меток классов объектов при упорядочивании по ним.

Коэффициент Джини

Изначально **коэффициент Джини (Gini coefficient[1])** использовался в экономике, социологии и статистике для оценки расслоения общества относительно какого-нибудь экономического показателя (чаще дохода). Визуально такое расслоение изображается с помощью **кривой Лоренца**, которая строится в координатах «процент населения» – «процент дохода».



Не путайте коэффициент Джини (Gini coefficient) с индексом Джини (Gini impurity), который используется в критерии расщепления при построении решающих деревьев.

Рис. XA.5. Кривая Лоренца для оценки расслоения общества.

На рис. XA.5 показана кривая Лоренца для населения из 5 человек с доходами: 1, 1, 2, 2, 3 (в некоторых условных единицах). Людей специально упорядочиваем по неубыванию дохода. Прохождение кривой через точку $(0.6, 4/9)$ говорит о том, что 60% населения (первые три человека) имеют $4/9$ от всех доходов $(1+1+2)$. Если бы доходы среди населения распределялись поровну, то кривая Лоренца проходила бы по диагонали. Неравномерность доходов оценивают коэффициентом Джини, который равен нормированной площади между линией равенства (диагональю) и кривой Лоренца:

$$\text{Gini} = \frac{A}{A+B} = 2A,$$

[1] Gini, C.: Variabilità e Mutuabilità. Contributo allo Studio delle Distribuzioni e delle Relazioni Statistiche. C. Cuppini, Bologna (1912).

т.к. $A+B=0.5$ – площадь «треугольной» половины квадрата. Если кривая проходит через точки $(p_0,i_0)=(0,0),(p_1,i_1),\ldots,(p_m,i_m)=(1,1)$, то

$$\text{Gini}=1-\sum_{t=1}^{m}(p_t-p_{t-1})(i_t+i_{t-1}).$$

В описанном примере $\text{Gini}=2/9$.

В машинном обучении кривая[1] Лоренца (она также иногда называется **CAP – Cumulative Accuracy Profile Curve**) строится немного по-другому – в координатах (PR,TPR), где PR = Positive Rate – доля объектов, которые при определённом выборе порога, отнесены к классу 1. На рис. XA.6 показана кривая Лоренца в примере, для которого мы раньше строили ROC-кривую (они немного похожи: кривая Лоренца «скошенный» аналог ROC-кривой).

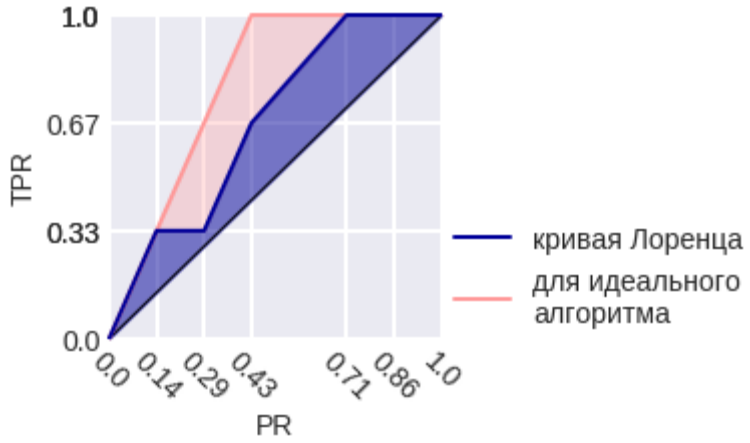


Рис. XA.6. Кривая Лоренца в машинном обучении.

Прохождение кривой Лоренца через точку $(2/7,1/3)$ означает, что когда при определённом выборе порога $(\theta=0.4)$ классификатор относит к классу 1 два объекта из семи, то среди них содержится треть объектов класса 1. Идеальная кривая Лоренца, с точки зрения классификации (когда объекты класса 1 имеют оценки выше, чем объекты класса 0) показана на рис. XA.6 розовым цветом. Коэффициент Джини здесь – отношение площади между кривой Лоренца и диагональю и площади между идеальной кривой Лоренца и диагональю. В данном случае – 7/12.

[1] Дмитрий Петухов Коэффициент Джини. Из экономики в машинное обучение. https://habr.com/ru/company/ods/blog/350440/

Оказывается, коэффициент Джини и показатель AUC_{ROC} связаны. Действительно, вспомним, что AUC_{ROC} это площадь по кривой в координатах (FPR, TPR), т.е.

$$\text{AUC}_{\text{ROC}} = \int_0^1 \text{TPR} \partial \text{FPR} = \int_0^1 \frac{\text{TP}}{m_1} \partial \frac{\text{FP}}{m_0} = \frac{1}{m_1 m_0} \int_0^{m_0} \text{TP} \partial \text{FP}$$

m_0 – число объектов класса 0 в выборке, m_1 – число объектов класса 1 в выборке. Очевидно, что

$$\text{Gini} = \frac{\int_0^1 \text{TPR} \partial \text{PR} - 0.5}{0.5 m_0 / (m_0 + m_1)} = \frac{\int_0^1 \frac{\text{TP}}{m_1} \partial \frac{\text{FP} + \text{TP}}{m_0 + m_1} - 0.5}{0.5 m_0 / (m_0 + m_1)},$$

в числителе здесь стоит разница между площадью под кривой в координатах (PR, TPR) и площадью нижнего треугольника (она равна половине площади единичного квадрата), в знаменателе – площадь закрашенного треугольника (синим и розовым цветом) с высотой равной 1 и основанием равным значению $1 - \text{PR}_*$, где PR_* соответствует порогу, при котором (в идеальном случае) оценки всех объектов класса 1 оказались выше порога, а оценки всех объектов класса 0 – ниже, т.е. доле объектов класса 1, т.е.

$$\text{PR}_* = \frac{m_1}{m_0 + m_1},$$

тогда

$$\text{Gini} = \frac{2(m_0 + m_1)}{m_1 m_0 (m_0 + m_1)} \int_0^m \text{TP} \partial (\text{FP} + \text{TP}) - \frac{(m_0 + m_1)}{m_0} =$$

$$= \frac{2}{m_1 m_0} \int_0^{m_0} \text{TP} \partial \text{FP} + \frac{2}{m_1 m_0} \int_0^{m_1} \text{TP} \partial \text{TP} - \frac{m_0 + m_1}{m_0} = 2\,\text{AUCROC} + \frac{\text{TP}^2}{m_1 m_0}\Big|_0^{m_1} - \frac{m_1}{m_0} - 1,$$

в результате получаем линейную связь между двумя популярными показателями качества:

$$\text{Gini} = 2\,\text{AUC}_{\text{ROC}} - 1.$$

Из формулы видно, что коэффициент Джини лежит на отрезке $[-1, +1]$, при этом «случайным» моделям соответствуют околонулевые значения.

Кривая Лоренца и **коэффициент Gini используются и в задачах регрессии**: пусть целевой признак – сумма страхового случая, тогда кривую Лоренца можно построить в координатах (PR, IR), где PR – доля объектов (случаев) с оценками алгоритма выше некоторого порога, IR – доля страховых выплат, которая приходится на эти объекты. Идеальный алгоритм здесь будет упорядочивать объекты по величине страхового случая. На рис. ХА.7 показана кривая Лоренца для решения с истинными целевыми значениями

$$5, 2, 10, 3, 0, 5, 0, 0$$

(при этом они идут в порядки убывания оценки, которую выдал алгоритм, т.е. последние два объекта правильно получили низкие оценки – у них нулевые выплаты, а вот первый объект не был правильно угадан – у него не самая большая выплата).

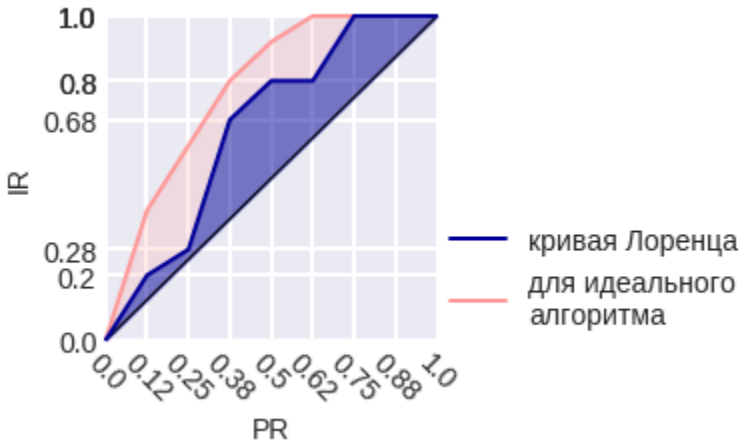


Рис. ХА.7. Кривая Лоренца в задаче регресии.

Прохождение кривой через точку (0.5, 0.8) означает, что на первую половину упорядоченного списка приходится 80% всех страховых выплат (5 + 2 + 10 + 3). Коэффициент Джини здесь равен примерно 0.57.

Кривая «полнота-точность»

На рис. ХА.8 показана кривая в координатах точность-полнота (PR-кривая) для модельной задачи из табл. 1, т.е. для разных бинаризаций оценок по порогу $\theta \in \mathbb{R}$ вычисляем полноту $R(\theta)$ и точность $P(\theta)$, соединяем их последовательно отрезками и получаем кривую. Площадь под ней называют **Average Precision** или $\mathbf{AUC_{PR}}$, её можно также интерпретировать как качество алгоритма, точнее вычисленных им оценок, т.е. опять это оценка семейства

бинарных классификаторов, которое порождено различными бинаризациями по порогу.

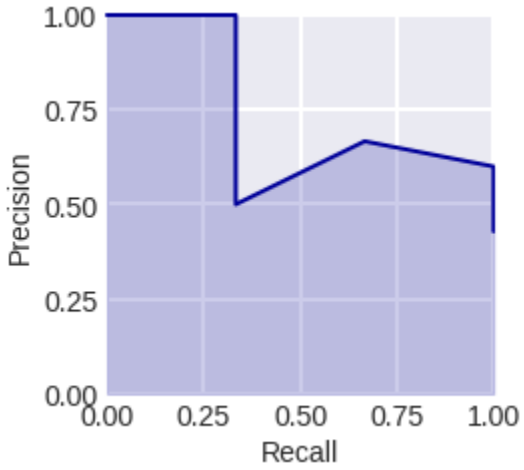


Рис. XA.8. Кривая полнота-точность для модельной задачи.

Точка $(1/3, 1/2)$ на PR-кривой рис. XA.8 соответствует порогу бинаризации ($\theta = 0.4$), при котором два объекта алгоритм относит к классу 1, из них 1 действительно прндлежит этому классу, поэтому $P = 1/2$, а всего объектов первого класса 3, поэтому $R = 1/3$.

PR-кривую часто используют в задаче с дисбалансом класса. С одной стороны это оправдано тем, что по ней можно выбрать порог для бинаризации исходя их двух ключевых показателей: точности и полноты. С другой стороны, как мы дальше покажем на примере, в отличие от ROC-кривой, PR-кривая неустойчива к дисбалансу, точнее прореживанию классов. Если при фиксированном пороге бинаризации удалить случайную часть объектов класса 1, то полнота не сильно изменится (т.к. это доля объектов с оценкой выше порога), а вот точность может измениться значительно (т.к. это доля объектов класса 1 среди объектов, оценка которых выше порога, а мы часть объектов удалили).

Другие кривые в задачах классификации

Опишем другие популярные кривые[1], которые строят для оценки качества классификации. **Кривую Лоренца**, т.е. кривую в координатах (PR, TPR) также часто называют **Gain-кривой (Gain Curve / Chart)** или **CAP (Cumulative**

[1] Реализации функций отрисовки некоторых кривых можно найти здесь: https://github.com/reiinakano/scikit-plot

Accuracy Profile)[1], иногда её называют также Lift Curve[2], хотя дальше мы представим ещё одну кривую с таким названием.

Далее будем показывать примеры кривых в задаче с линейными плотностями[3]. На рис. XA.9 показана «теоретическая[4]» Gain-кривая (синим) и эмпирические (тонкими красными линиями), вычисленные по конечным выборкам, которые сгенерированы с указанными плотностями. Рассмотрим «**задачу о предложении услуги**»: необходимо сделать классификатор, который по описанию клиента предсказывает, откликнется ли он на предложение услуги (класс 1 – это клиенты, которые откликнутся, класс 0 – все остальные). Обзвон клиентов с предложением такой услуги лучше делать по потенциально откликающимся клиентам, так будет меньше затрат на звонки, экономия времени операторов и меньше недовольных спамом клиентов. Gain-кривая как раз показывает зависимость TPR (доли заключённых договоров среди всего множества договоров, которые могли быть заключены – клиент согласился бы, если бы ему предложили) от PR (числа обзваниваемых по алгоритму клиентов – мы выбираем лишь тех, у кого оценка выше некоторого порога).

Gain-кривая показывает покрытие целевой аудитории от масштаба контакта.

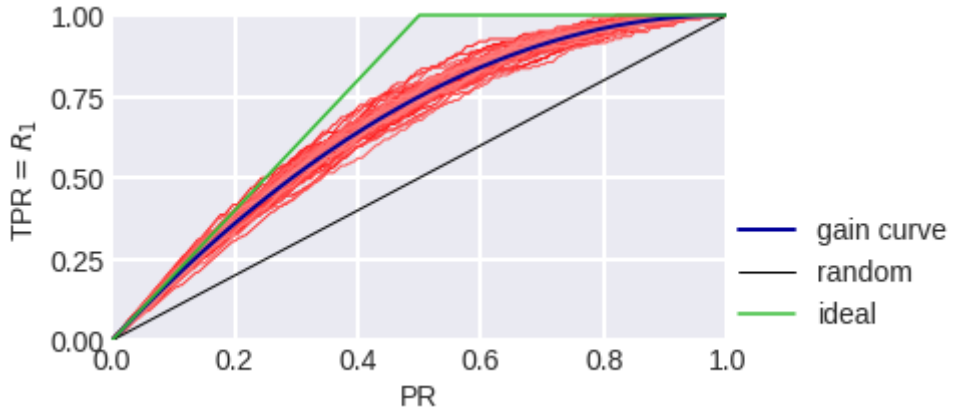


Рис. XA.9. Gain-кривые в модельной задаче: синяя – теоретическая, красные – эмпирические по выборке из 3000 объектов.

В описанной интерпретации понятно, почему случайному алгоритму на рис. XA.9 соответствует чёрная диагональ. Если случайно обзвонить долю p

[1] Řezáč M., Řezáč F. How to measure the quality of credit scoring models //Finance a úvěr: Czech Journal of Economics and Finance. – 2011. – Т. 61. – №. 5. – С. 486-507. (но здесь кривая TP(PR))

[2] Vuk M., Curk T. ROC curve, lift chart and calibration plot //Advances in methodology and Statistics. – 2006. – Т. 3. – №. 1. – С. 89–108-89–108.

[3] См. главу про показатели качества в задаче чёткой бинарной классификации.

[4] Для «бесконечной» выборки.

клиентов, то среди клиентов класса 1 в обзвон попадёт доля p. Чем выше расположена Gain-кривая относительно диагонали, тем лучше.

Отношение высот Gain-кривой и диагонали часто изображают в виде **Lift-кривой (Lift Curve / Chart)**: она строится в координатах $(\text{PR}, \text{TPR}/\text{PR})$. На рис.XA.10 показаны Lift-кривые, соответствующие нарисованным выше Gain-кривым.

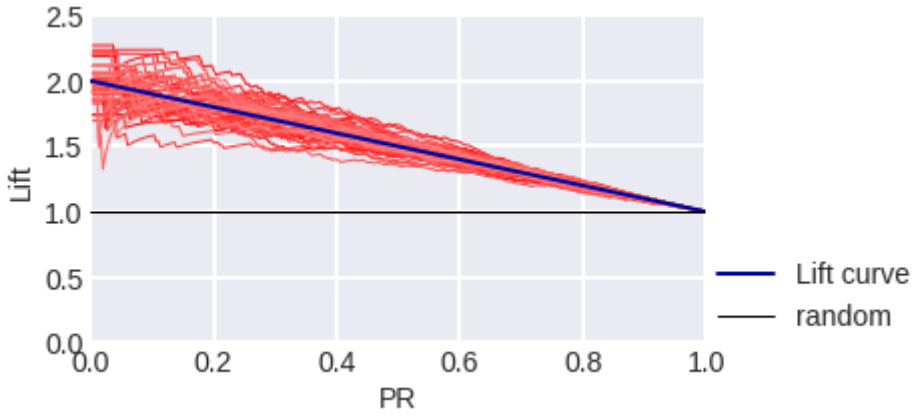


Рис. XA.10. Lift-кривые в модельной задаче: синяя – теоретическая, красные – эмпирические по выборке из 3000 объектов.

В банковской среде приняты термины типа **Gain-Top-10%** или **Lift-Top-10%**, это просто значения TPR или TPR/PR, когда 10% объектов с наивысшими оценками алгоритм относит к классу 1 (т.е. при $\text{PR} = 0.1$). Также принято строить эти кривые лишь по точкам

$$\text{PR} = 0, 0.1, 0.2, \ldots, 1.$$

Ранее было доказано, что

$$\text{Gini} = \frac{\int_0^1 \text{TPR}\,\partial\,\text{PR} - 0.5}{0.5 m_0 / (m_0 + m_1)} = 2\,\text{AUC}_{\text{ROC}} - 1,$$

отсюда получаем площадь под Gain-кривой

$$\int_0^1 \text{TPR}\,\partial\,\text{PR} = \frac{1}{2}\frac{m_0}{m_0 + m_1}(2\,\text{AUC}_{\text{ROC}} - 1) + \frac{1}{2} =$$

$$= \frac{1}{m_0 + m_1}\left(m_0\,\text{AUC}_{\text{ROC}} + \frac{m_1}{2} \right).$$

По смыслу это **вероятность того, что у случайного позитивного объекта оценка выше, чем у случайного**.

Kolomogorov-Smirnov (K-S) chart используется для сравнения распределений объектов классов 1 и объектов классов 0 на оси PR[1]. Строятся две кривые: $\mathrm{TPR}(\mathrm{PR})$ и $\mathrm{FPR}(\mathrm{PR})$. Первая, кстати, знакомая нам кривая Лоренца / Gain-кривая: доля объектов класса 1, которые алгоритм отнёс к классу 1, в зависимости от доли объектов, которые алгоритм отнёс к классу 1. Смысл второй – доля объектов класса 0, которые алгоритм отнёс к классу 1, в зависимости от доли объектов, которые алгоритм отнёс к классу 1. На рис. XA.11 показаны соответствующие кривые для модельной задачи с линейными плотностями. Максимальная разница

$$\max_{\mathrm{PR}}(\mathrm{TPR}-\mathrm{FPR})$$

между кривыми часто называется **KS-расстоянием**. Как мы уже отмечали, значения $\mathrm{TPR}(\theta)$ и $\mathrm{FPR}(\theta)$ почти не зависят от баланса, т.к. устойчивы к «прореживанию классов», а $\mathrm{TPR}(\mathrm{PR})$ и $\mathrm{FPR}(\mathrm{PR})$ меняются при изменении баланса, но KS-расстояние почти не меняется.

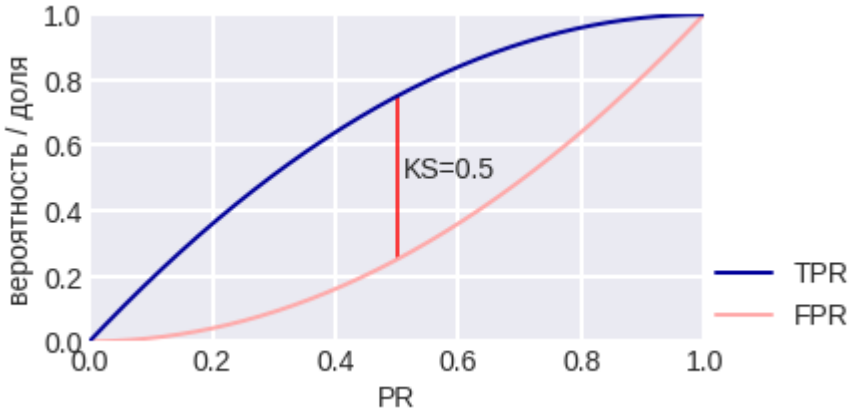


Рис. XA.11. K-S chart для модельной задачи

При **анализе доходности (Profit Analysis)** обычно **используют таблицу выгоды (The Gains Table)**, для её построения объекты упорядочиваются по убыванию оценки принадлежности к классу 1, которую выдал алгоритм, потом разбиваются на 10 равных частей – децилей, каждому децилю соответствует строка таблицы, см. табл.:

- N – число объектов в дециле,

[1] важно: а не по оценкам, которые выдаёт алгоритм.

- % – процент объектов в дециле,
- cum_... – кумулятивное значение, например cum_% – сколько процентов объектов до этого дециля включительно,
- Prob – доля объектов из класса 1 в дециле,
- N_t – число объектов из класса t,
- %_t – какой процент объектов класса t попал в дециль,
- K-S – разница распределений по Колмогорову-Смирнову: cum_%1 – cum%_0, Lift – отношение cum_%1 / %.

N	%	cum_%	Prob	N_1	%_1	cum_N1	cum_%1	N_0	%_0	cum_N0	cum_%0	K-S	Lift
11238	10.0%	10.0%	0.229	2572	49.0%	2572	49.0%	8666	8.1%	8666	8.1%	40.9%	4.902
11237	10.0%	20.0%	0.081	912	17.4%	3484	66.4%	10325	9.6%	18991	17.7%	48.7%	3.320
11238	10.0%	30.0%	0.050	565	10.8%	4049	77.2%	10673	10.0%	29664	27.7%	49.5%	2.572
11237	10.0%	40.0%	0.037	413	7.9%	4462	85.0%	10824	10.1%	40488	37.8%	47.2%	2.126
11238	10.0%	50.0%	0.025	282	5.4%	4744	90.4%	10956	10.2%	51444	48.0%	42.4%	1.808
11237	10.0%	60.0%	0.018	197	3.8%	4941	94.2%	11040	10.3%	62484	58.3%	35.8%	1.569
11237	10.0%	70.0%	0.013	146	2.8%	5087	97.0%	11091	10.4%	73575	68.7%	28.3%	1.385
11238	10.0%	80.0%	0.008	94	1.8%	5181	98.7%	11144	10.4%	84719	79.1%	19.7%	1.234
11237	10.0%	90.0%	0.005	51	1.0%	5232	99.7%	11186	10.4%	95905	89.5%	10.2%	1.108
11238	10.0%	100.0%	0.001	15	0.3%	5247	100.0%	11223	10.5%	107128	100.0%	0.0%	1.000

Табл. Статистика доходности.

По таблице можно посчитать экономику, связанную с задачей. Например, если таблица соответствует описанной выше задаче предложения услуги, стоимость контакта с клиентом равна 1$, а доход с отклика равен 5$, тогда если проконтактировать с 10% клиентов, то

общие траты на контакты = 11 238$,

потенциальный доход = 2572*5$ = 12 860$,

прибыль = 12 860$ – 11 238$ = 1 622$.

Заметим, что контактировать с 20% клиентов уже не выгодно:

общие траты на контакты = 22 475$,

потенциальный доход = 3484*5$= 17 420$,

прибыль = 17 420$ – 22 475$ < 0.

Приложения и примеры

1. Приведём **пример построения ROC-кривой и вычисления площади в модельной задаче**. Используем уже знакомую задачу: на оценках $b(x) \in [0,1]$, которые получает алгоритм, объекты класса 0 распределены с плотностью $p_0(b) = 2 - 2b$, а объекты класса 1 – с плотностью $p_1(b) = 2b$, см. рис. XX.7. Значение TPR при выборе порога бинаризации θ равно

$$\mathrm{TPR}(x) = 1 - \theta^2,$$

а FPR равно

$$\mathrm{FPR}(x) = (1-\theta)^2.$$

Параметрическое уравнение для ROC-кривой получено, можно уже сразу вычислить площадь под ней:

$$\int_1^0 \mathrm{TPR}(\theta) \cdot \mathrm{FPR}'(\theta) \partial\theta = 2\int_0^1 (1-\theta^2)(1-\theta)\partial x$$

или выразить TPR через FPR:

$$\mathrm{TPR} = 2\sqrt{\mathrm{FPR}} - \mathrm{FPR},$$

Плотности линейные, а ROC-кривая даже не полиномиальная функция.

тогда площадь под ROC-кривой

$$\int_0^1 (2\sqrt{t} - t)\partial t = \frac{5}{6} \approx 0.83.$$

Заметим, что максимальная точность достигается при пороге бинаризации 0.5 и равна 3/4 = 0.75 (что не кажется очень большой в решаемой сбалансированной задаче). Это частая ситуация: **$\mathbf{AUC_{ROC}}$ существенно выше максимальной достижимой точности (accuracy[1])**! Отметим, что **функционал $\mathbf{AUC_{ROC}}$ оценивает набор полученных оценок** (в некотором смысле, целое семейство классификаторов для разных порогов

AUC_{ROC} «завышает качество».

[1] Кстати, AUC ROC бинаризованного решения (при пороге бинаризации 0.5) равна 0.75! Подумайте, почему это значение совпало с точностью?

бинаризации), **а не конкретный классификатор** (ему соответствует лишь точка на ROC-кривой).

В такой «непрерывной» постановке задачи (когда объекты двух классов описываются плотностями) AUC ROC имеет вероятностный смысл: это **вероятность того, что случайно взятый объект класса 1 имеет оценку принадлежности к классу 1 выше, чем случайно взятый объект класса 0**:

$$\text{AUC}_{\text{ROC}} = \mathbf{P}(b(x_i) < b(x_j) \mid y_i = 0, y_j = 1).$$

В «дискретном» случае (для конечной выборки) вероятность превращается в долю, см. рис. XA.3, которая является оценкой вероятности. Посмотрим, как эта оценка стремится к вероятности при увеличении объёма выборки. Для нашей модельной задачи возьмём конечные выборки разной мощности с указанными распределениями. На рис. XA.12 показаны значения AUC_{ROC} в таких экспериментах: все они распределены около теоретического значения 5/6, но разброс достаточно велик для небольших выборок.

Для оценки AUC ROC выборка в несколько сотен объектов мала.

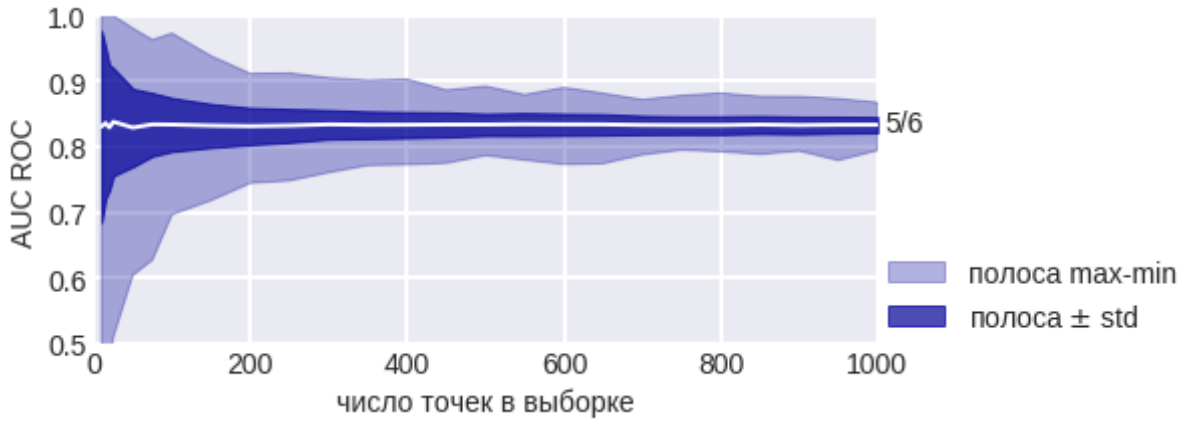


Рис. XA.12. Варьирование AUC_{ROC} в экспериментах.

Также полезно посмотреть, как выглядят ROC-кривые в наших экспериментах, см. рис. XA.13. Естественно, при увеличении объёма выборок ROC-кривые, построенные по выборкам, будут сходиться к теоретической кривой (построенной для распределений).

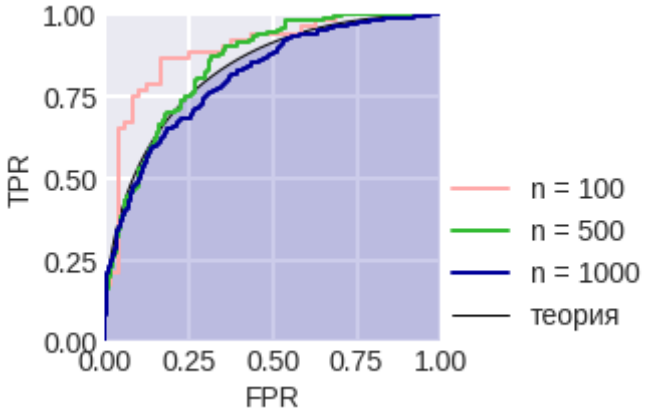


Рис. XA.13. ROC-кривые в экспериментах.

2. Площади под многими описанными кривыми можно **использовать для оценки качества признаков**. Классический показатель качества признака – коэффициент корреляции признака с целевым признаком, хотя он оценивает лишь линейную зависимость. Если считать, что значения признака – это ответы алгоритма (не обязательно они должны быть нормированы на отрезок $[0,1]$, ведь нам важен порядок), тогда выражение

$$2\cdot|\,\mathrm{AUC}_{\mathrm{ROC}}-0.5\,|$$

вполне подойдёт для оценки качества признака: оно максимально, если по этому признаку 2 класса строго разделяются и минимально, если они «перемешаны». Такой подход позволяет находить монотонные зависимости целевых значений от значений признака, а не только линейные, но не находит унимодальные зависимости, например когда метки объектов, упорядоченных по значению признака выглядят так:

$$(0,\ldots,0,1,\ldots,1,0,\ldots,0)\,.$$

3. Часто утверждается, что показатель $\mathrm{AUC}_{\mathrm{ROC}}$ не годится для **задач с сильным дисбалансом классов**. Рассмотрим одно из обоснований этого. Пусть в задаче информационного поиска 1 000 000 объектов (сайтов в интернете), при этом только 10 объектов из класса 1 (сайты, релевантные некоторому запросу). Рассмотрим алгоритм, ранжирующий все сайты в соответствии с этим запросом (по этому ранжированию формируется выдача на поисковый запрос). Пусть он в начало списка поставил 100 объектов класса 0, потом 10 – класса 1, потом – все остальные класса 0, см. рис. XA.14 (слева). Значение $\mathrm{AUC}_{\mathrm{ROC}}$ будет довольно высоким: 0.9999, но ответ алгоритма (выдачу поисковика) нельзя считать хорошей: в верхней части выдачи 100 нерелевантных сайтов (обычно пользователь смотрит топ выдачи, иногда

пролистывает несколько страниц, вряд ли он доберётся до 10 релевантных ссылок).

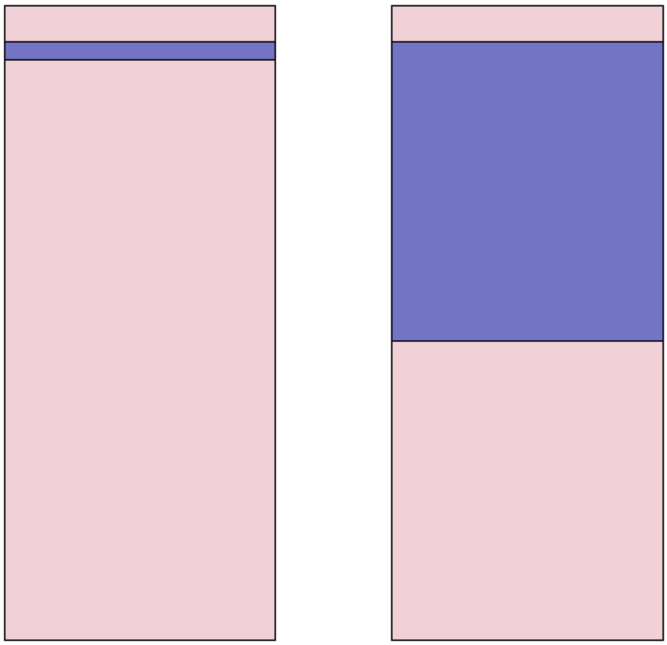


Рис. ХА.14. Объекты класса 1 (релевантные) и класс 0 (нерелевантные), упорядоченные по уменьшению оценки за класс 1, слева – при дисбалансе, справа – при балансе классов.

Некорректность такого обоснования, во-первых, в логической ошибке: приведён один пример задачи с дисбалансом классов, в которой показатель AUC_{ROC} принимает большие значения при неудовлетворительном ответе. Отсюда не следует, что во всех задачах с дисбалансом будет похожая ситуация. Заметим, что упорядочивание немного искусственное, обычно большие оценки получают объекты с ярко выраженными паттернами класса 1.

Во-вторых, можно привести аналогичные пример для сбалансированных классов. Пусть в рассмотренном примере объектов класса 1 могло быть 500 000 – ровно половина, тогда показатель AUC_{ROC} будет чуть меньше: 0.9998, см. рис. ХА.14 (справа), но всё равно очень большим. И также упорядочивание, полученное алгоритмом, нельзя назвать приемлемым: большие оценки получили нерелевантные объекты.

Таким образом, исходный пример не использует дисбаланс классов, в нём используется специальная постановка задачи информационного поиска, в которой неявно предполагается, что большие оценки должны получать релевантные объекты (релевантные объекты должны быть вверху поисковой выдачи). **В задачах поиска показатель AUC_{ROC} действительно не применим.**

Для таких задач есть другие функционалы качества, кроме того, есть специальные вариации AUC, например AUC@k[1].

4. **В банковском скоринге** показатели Джини и AUC_{ROC} (они эквивалентны с точностью до линейного преобразования) довольно популярны, хотя очевидно, что они не особо подходят для подобных задач. Банк может выдать ограниченное число кредитов, поэтому главное требование к алгоритму – чтобы среди объектов, которые получили наименьшие оценки были только представители класса 0 («вернёт кредит», если мы считаем, что класс 1 – «не вернёт»). Об этом можно судить по форме ROC-кривой, см. рис. XA.15: слева из-за того, что группа объектов с низкими оценками принадлежности к классу 1 имеет истинные метки равные 0, кривая прижимается к прямой $TPR = 1$ в верхнем правом углу. Аналогично, справа кривая прижимается к $FPR = 0$, поскольку группа объектов с высокими оценками имеет метку равную 1. Заметим, что площадь под указанными кривыми небольшая (не сильно больше 0.5).

В одном проекте была задача построить алгоритм, который по действия пользователя на сайте авиакомпании предсказывает, купит ли он билет или покинет сайт. Получить решение с большим значением AUC_{ROC} не удалось, но зато алгоритм практически безошибочно угадывал группу посетителей сайта, которые не будут покупать (ROC-кривая выглядела как на рис. XA.15 слева). Таким посетителям логично предложить билеты со скидкой, чтобы всё-таки побудить их к покупке.

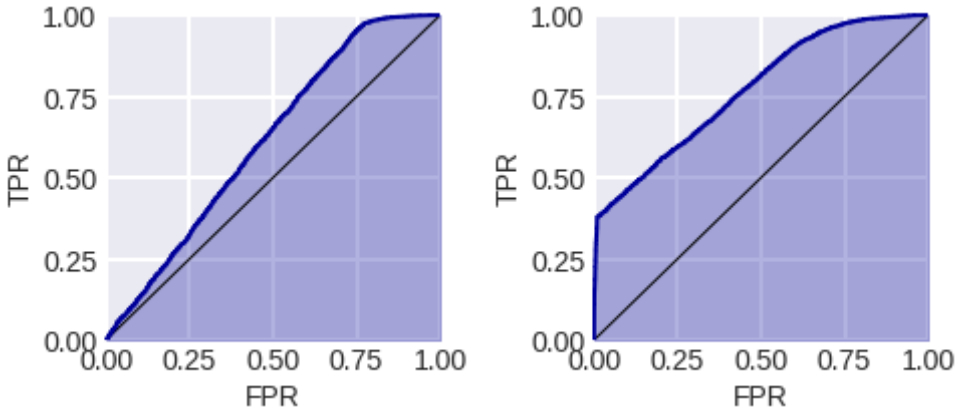


Рис. XA.15. Форма ROC-кривой при верных ответах с низкими оценками (слева) и с высокими (справа).

[1] Rosenfeld N. et al. Learning structured models with the AUC loss and its generalizations //Artificial Intelligence and Statistics. – PMLR, 2014. – С. 841-849.

5. Рассмотрим, как выглядит **PR-кривая в задаче с линейными плотностями**. Ранее были выведены формулы для полноты и точности от порога бинаризации:

$$R = 1 - \theta^2,\ P = (1+\theta)/2,$$

из которых следует уравнение для PR-кривой:

$$P = \frac{1+\sqrt{1-R}}{2}.$$

На рис. XA.16 такая кривая показана (синим цветом), а также кривые построенные по конечным случайным выборкам, которые распределены с линейными плотностями (тонкие красные линии). Слева – для «небольших» выборок (из 300 объектов), справа – для «больших» выборок (из 3000 объектов). Заметим, что в общем случае PR-кривая не выпуклая[1]. В нашей модельной задаче площадь AUC_PR равна[2]

$$\mathrm{AUC}_{\mathrm{PR}} = \int_0^1 P\,\partial R = \int_0^1 \frac{1+\sqrt{1-R}}{2}\,\partial R = \frac{5}{6} = 0.83(3).$$

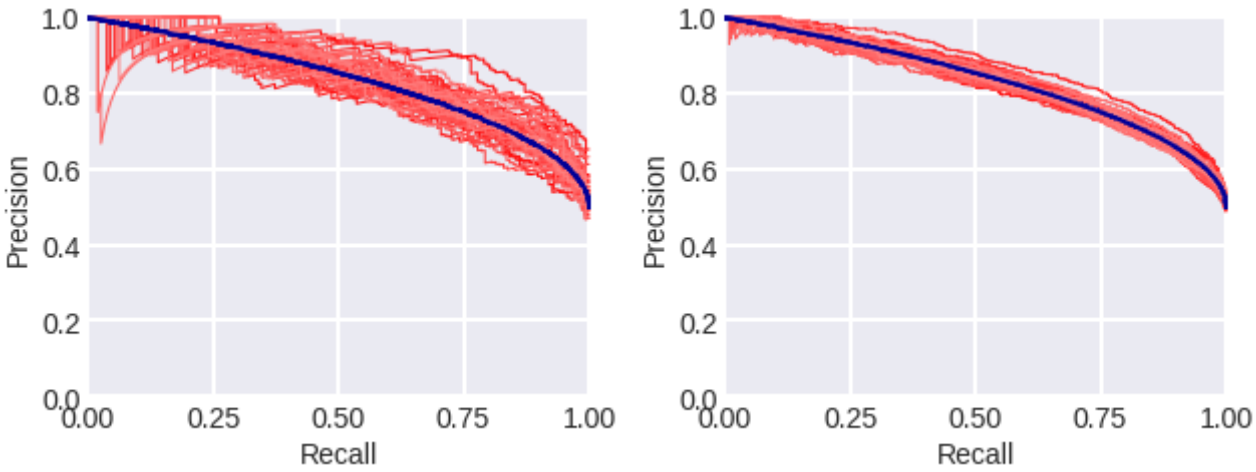


Рис. XA.16. Теоретические (синие) и эмпирические (красные) PR-кривые, построенные по выборкам из 300 (слева) и 3000 (справа) объектов.

6. Поговорим про **площадь под PR-кривой при дисбалансе классов.** В задаче с линейными плотностями можно сделать дисбаланс: разную пропорцию классов. На рис. XA.17 показано, как эмпирическая оценка $\mathrm{AUC}_{\mathrm{PR}}$ зависит от объёма выборки при разной пропорции классов: когда классы равновероятны и

[1] ROC-кривая тоже.

[2] Для выборок из 300 объектов после 100 экспериментов 0.839 ± 0.024(std), для выборок из 3000 – 0.833 ± 0.012 (std).

когда есть дисбаланс классов (представителей класса 1 в 9 раз больше, чем класса 0). Светлым цветным коридором показаны стандартные отклонения от среднего. Видно, что для задачи с дисбалансом классов они больше.

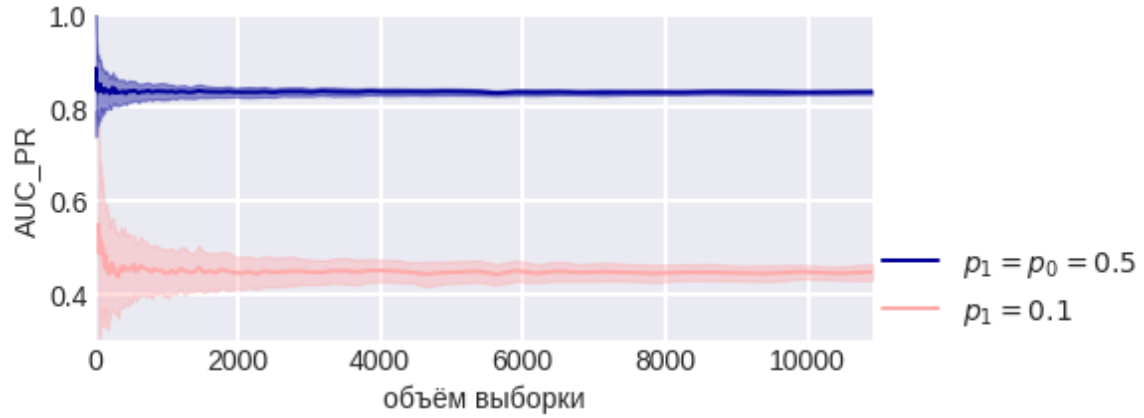


Рис. XA.17. Оценка AUC_PR для разного объёма выборки и баланса классов.

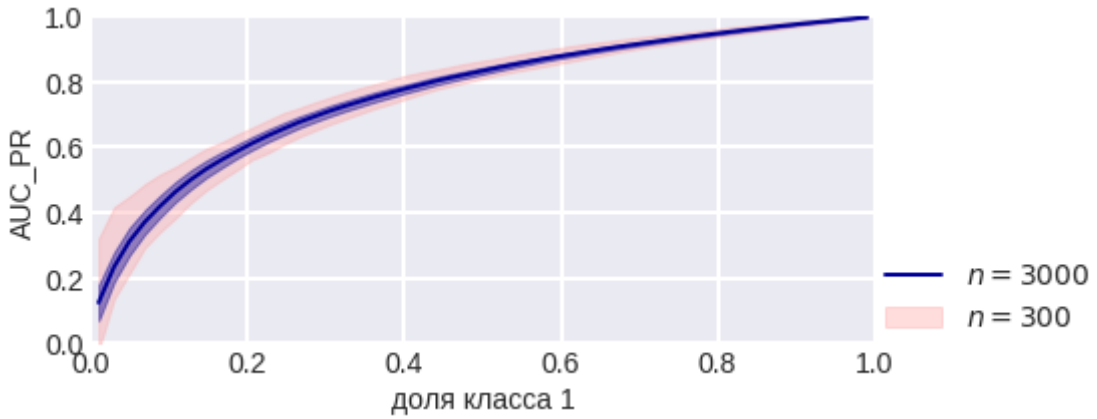


Рис. XA.18. Оценка AUC_{PR} для разного баланса классов.

Самое интересное, что при внесении дисбаланса площадь уменьшилась. Можно проанализировать, как она зависит от пропорции классов, см. рис. XA.18. Получается, что **площадь под PR-кривой зависит от пропорции классов**[1] и при уменьшении доли объектов класса 1 (например, при «прореживании» объектов класса 1 – удалении случайного подмножества объектов класса 1) показатель стремится к 0. Это делает **практически невозможным интерпретировать значение AUC_{PR}, в отрыве от знания особенностей задачи**. Например, фраза «$AUC_{PR} = 0.4$» может говорить об очень хорошем качестве решения задачи при сильном дисбалансе классов, при этом в случае сбалансированной выборки это достаточно низкий показатель качества. Тем не менее, площадь под PR-кривой часто рекомендуют использовать в задачах с дисбалансом классов, аргументируя это тем, что PR-кривая точнее описывает

[1] Напомним, что ROC-кривая не зависит.

правильность классификации объектов с большими оценками, тогда как ROC-кривая – различие распределений объектов разных классов по оценкам[1].

[1] Подумайте, корректна ли такая аргументация? Как быть с увеличением погрешности при оценке площади под PR-кривой в задачах с дисбалансом?

Задачи и вопросы

1. Докажите, что в задаче с линейными плотностями при доле объектов первого класса $p_1 = 0.1$ максимум $\mathrm{TPR} - \mathrm{FPR}$ равен 0.5 и достигается при $\mathrm{PR} = 0.3$. Соответствует ли такому случаю рис. XA.19? Какому порогу бинаризации (каким свойствами он обладает) соответствует максимум TPR – FPR? Верно ли, что значение $\mathrm{TPR} - \mathrm{FPR}$ не зависит от баланса классов (не меняется при прореживании одного из классов)?

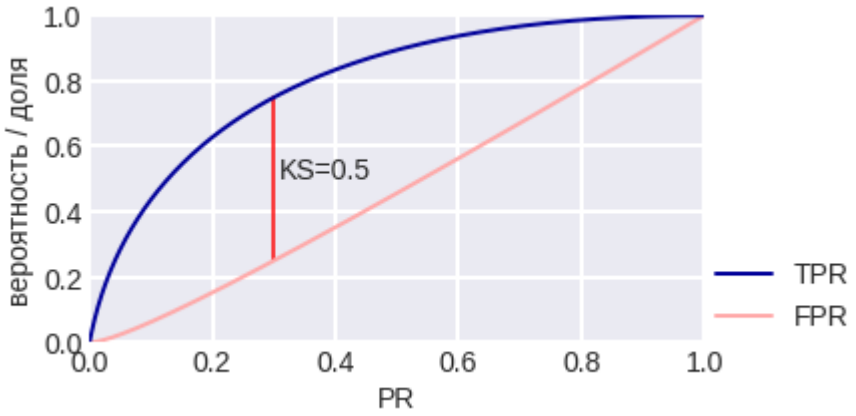


Рис. XA.19. K-S chart для модельной задачи с дисбалансом классов

2. Для оптимизации площади под ROC-кривой было предложено перейти к задаче с парами объектов. Как по классификатору пар вычислить оценки принадлежности к классу 1 исходных объектов?

3. Могут ли в задаче с дисбалансом теоретические кривые (вычислены по «бесконечным выборкам») и эмпирические (по конечным случайным выборкам) выглядеть как на рис. XA.20-21. Например, могут ли эмпирические Gain-кривые располагаться в основном выше теоретической кривой?

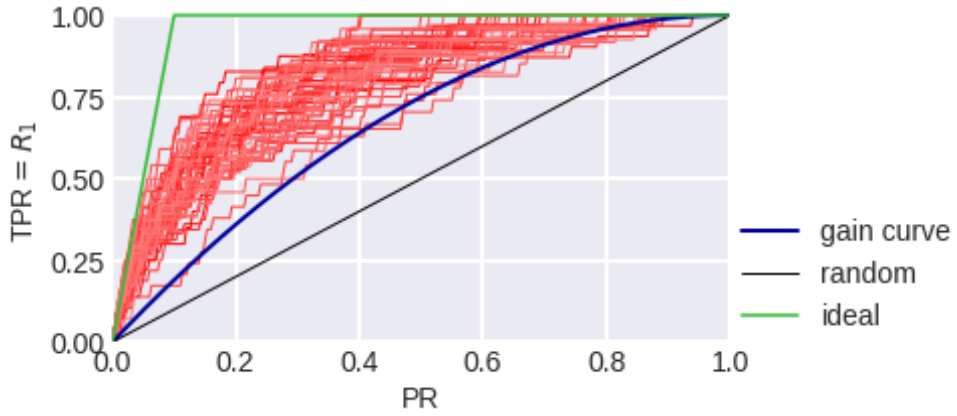


Рис. XA.20. Gain-кривые в модельной задаче с дисбалансом классов: синяя – теоретическая, красные – эмпирические по выборке из 300 объектов.

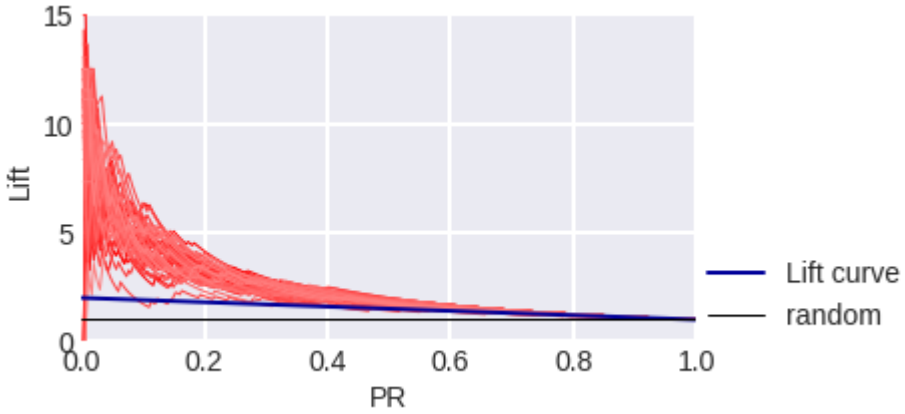


Рис. XA.21. Lift-кривые в модельной задаче с дисбалансом классов: синяя – теоретическая, красные – эмпирические по выборке из 300 объектов.

4. Известно, что что $\text{AUC}_{\text{ROC}} < 1$ на выборке из m объектов. Какое максимальное значение может быть у AUC_{ROC}?

5. Может ли усреднение ответов двух алгоритмов с $\text{AUC}_{\text{ROC}} = 1$ иметь $\text{AUC}_{\text{ROC}} < 1$ (на фиксированной выборке)? Может ли усреднение ответов двух алгоритмов с $\text{AUC}_{\text{ROC}} = 0.5$ иметь $\text{AUC}_{\text{ROC}} = 1$?

6. Над оценками алгоритма $b = b(x)$ производят следующие операции:

- $\text{round}(b, 2)$ – округление до второго знака после запятой,
- $\max(b, 0.5)$,
- $(b+1)/2$,

как будет меняться (увеличиваться, уменьшаться, не изменяться) площадь под ROC-кривой после каждой из них? Если показатель изменяется, на сколько максимально он может измениться?

7. Как изменятся точность при прореживании одного из классов?

8. Получите необходимое и достаточное условие для кривой быть ROC-кривой для некоторой конечной размеченной выборке и набора оценок.

AUC ROC: итоги

- площади под ROC-кривой и PR-кривой не зависят от самих значений оценок, а только от того как упорядочены объекты выборки по невозрастанию этих оценок,
- следствием из предыдущего свойства получаем, что эти показатели не меняются при строго возрастающем преобразовании оценок,
- AUC_{ROC} не зависит от баланса классов (не меняется при прореживании классов), AUC_{PR} – зависит.
- некоторые показатели (AUC_{ROC}, AUC_{PR}) могут использоваться для оценки важности признаков,
- у площадей под кривыми неинтуитивная интерпретация для неспециалиста, например у AUC_{ROC} – вероятность правильного упорядочивания объектов (если взять случайную пару объектов из разных классов, то оценка объекта класса 1 больше оценки объекта класса 0),
- площади под кривыми оценивают не конкретный классификатор, а набор оценок; AUC_{ROC} «завышает качество»,
- коэффициент Джини линейно связан с показателем AUC_{ROC}, имеет интересное обобщение на задачу регрессии,
- популярные кривые в машинном обучении: ROC-кривая, PR-кривая, кривая Лоренца (она же Gain-кривая), Lift-кривая, K-S chart,
- по таблице выгоды можно оценить рентабельность некоторых проектов с машинным обучением.

Спасибо за внимание к книге!
Замечания по содержанию, замеченные ошибки и неточности можно написать в телеграм-чате
https://t.me/Dyakonovsbook